Short Answer:

Answer the following questions with complete sentences in your own words. You are encouraged to conduct your own research online or through other methods before answering the questions. If you research online, please consult multiple sources before you write down your answers. You are expected to be able to explain your answers in detail (Provide examples for each question).

1. What is System.IO? What is it used for?
2. What is a stream in C#?
3. What is the difference between the File class and FileStream class?
4. What does the following code do:

```
File.WriteAllText(@"C:\ChristmasMusic\Playlist.txt", "This is a
playlist for Christmas Hits");
```

5. What are serialization and deserialization
6. C# exceptions occur in runtime or compile time?
7. What is a user-defined Exception and why do we need to use a user-defined Exception? How to create one?
8. Is there anything wrong with the following exception handler as written? Will this code compile?

```
try { }
catch (Exception e) { }
catch (DivideByZeroException a) { }
```

9. What is the Exception filter?
10. What is a delegate?
11. What is the common usage of delegate?
12. What are the built-in generic delegates?
13. Define a delegate that points to the following methods:

```
public static string ToNumString (int num)
{
  string returnStr = $"Number {num}";
  return returnStr;
}
```

14. What is a multicast delegate?
15. Explain lambda expression, what does it do?

Coding Questions:

Write code in c# to solve the following problems. Please write your own answers. You are highly encouraged to present more than one way to answer the questions. Please follow best practices when you write the code so that it is easily readable, maintainable, and efficient. Clearly state your assumptions if you have any. You may discuss with others on the questions, but please write your own code.
(For your Collections HW please try to finish them as much as you can, I've already extended the DDL to Next Monday)

1. Read from the Holiday.txt and count the number of occurrences of each word(use Dictionary).

2. Create a user-defined exception class called NonIntResultException which is generated when the result of dividing two integers values produces a result with a fractional component [i.e the result is not an integer]
• NonIntResultException contains:
• Generates an appropriate message, for example, if the two integers are 7 and 2 . the resulting exception message would be "7 divided by 2 is not an integer"

3. Create the IntegerArrayMath class with int division method:
A. Loops thru instance field array and attempts to divide each value of the number array by the corresponding value of denom instance field array. such as number[0]/denom[0] and number[1]/denom[1],etc
B. If the result of the division is an integer then print out a message indicating the result of the division such as 8/4 is 2.
C. If the result of the division is not an integer then throw and handle a NonIntResultException and continue processing the result of the number array elements.
D. The method should use exception handling and also handle any attempt to divide by zero(arithmetic exception) the program should display an appropriate message and then continue processing the rest of the number array elements
E. Assume both arrays are the same length.

```
Examples:
Input:
Number is [4,8,15] and denom is [2,0,4]
Output:
The resultant output would be: 4/2 is 2
Division by zero is undefined
result 15 divided by 4 is not an integer
```

4. Use a built-in generic delegate to extract the if-else condition and pass it as a parameter to the function. We also encourage you to practice passing delegate as method parameters using your own examples.

```
// Output the result based on the student's score
0 references
public void PassExamWithoutDelegate (Student s)
{
    if (s.Score >= 60)
    {
        Console.WriteLine("Pass! Score = {0}", s.Score);
    }
    else
    {
        Console.WriteLine("Fail! Score = {0}", s.Score);
    }


}
```

Here is the Student Class:

```
public class Student
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public int Score { get; set; }
    }
```